

Einführung in Rechnernetze Sommersemester 2017

2. Übungsblatt

Aufgabe 1: HTTP-Quiz

Richtig oder falsch?

- Ein Nutzer fragt nach einer Webseite, die aus Text und drei Bildern besteht. Der Nutzer sendet eine Request-Nachricht und erhält darauf vier Response-Nachrichten.
- Zwei verschiedene Webseiten (z.B. `http://telematics.tm.kit.edu/staff.php` und `http://telematics.tm.kit.edu/ss2017_pse.php`) können über dieselbe persistente Verbindung gesendet werden.
- Mit nicht-persistenten Verbindungen können mehrere Objekte zwischen einem Webbrowser und einem Webserver über dieselbe TCP-Verbindung gesendet werden.
- Das `Date:-`Header-Field in einer HTTP-Response-Nachricht zeigt an, wann die Datei zuletzt geändert wurde.
- HTTP-Response-Nachrichten können keinen leeren Body haben.

Aufgabe 2: DNS in der Praxis

Mithilfe des Linux/Unix-Tools `dig` lassen sich DNS-Anfragen versenden. Mittels des Kommandos `dig @128.8.76.3 umd.edu A` können Sie zum Beispiel am Name-Server 128.8.76.3 nach dem A-Record für `umd.edu` fragen. Die Angabe eines Name-Servers ist dabei optional: Mittels `dig umd.edu A` wird die Anfrage an Ihren lokalen Name-Server gesendet.

- (a) Ermitteln Sie aus Ihren Netzeinstellungen Ihren lokalen Name-Server (Achtung: Dies ist nicht localhost!)
- (b) Wie lautet die IP-Adresse von `www.kit.edu`? Fragen Sie sowohl nach der IPv4- als auch nach der IPv6-Adresse!
- (c) Ermitteln Sie die für `uka.de` zuständigen Name-Server!

- (d) Wie lauten die IP-Adressen der Mail-Server, die für Ihre @student.kit.edu-Adresse verantwortlich sind?
- (e) Wählen Sie von den DNS-Root-Servern (<http://root-servers.org>) einen aus. Arbeiten Sie sich durch die Domänenhierarchie von `telematics.tm.kit.edu`, um die IP-Adresse(n) dieses Servers zu ermitteln. Gehen Sie dabei iterativ vor, und stellen Sie die Anfragen immer nur an die zuständigen Name-Server!

Aufgabe 3: CDN, DNS, und DASH

Sie gründen das Startup HuxVideo, welches Nutzer mit Video-on-Demand versorgen soll. Ihr Verkaufsargument ist die Anonymität der Nutzer, indem auf Nutzeraccounts, Tracking und ähnliches verzichtet wird. Das bedeutet, dass alle Inhalte jederzeit ohne Login zugänglich sind.

In Ihrem Rechenzentrum platzieren Sie deshalb drei Dienste: Einen Webserver, welcher den Nutzer die Inhalte über den Webbrowser durchsuchen lässt, einen Content-Server, welche alle Videochunks beinhaltet, und einen Name-Server, welcher Ihre neu gekaufte Domäne `huxvideo.com` verwalten soll. Das Streaming erfolgt via DASH über einen browserbasierten Player, der ebenfalls vom Webserver ausgeliefert wird. Abbildung 1 zeigt die entsprechenden Server inklusive ihrer Hostnamen. Zunächst entscheiden Sie sich gegen die Verwendung eines CDN, da Sie noch keine Kunden haben.

Um Ladezeiten zu optimieren, binden Sie alle Bilder, Stylesheets und Skripte direkt in Ihre HTML-Dokumente ein, sodass außer den Videoinhalten nur ein HTTP-Aufruf pro Seite auf Ihrem Webserver notwendig ist. Desweiteren kennt Ihr Webserver alle Inhalte des Content-Servers, sodass keine Kommunikation zwischen diesen beiden Komponenten notwendig ist.

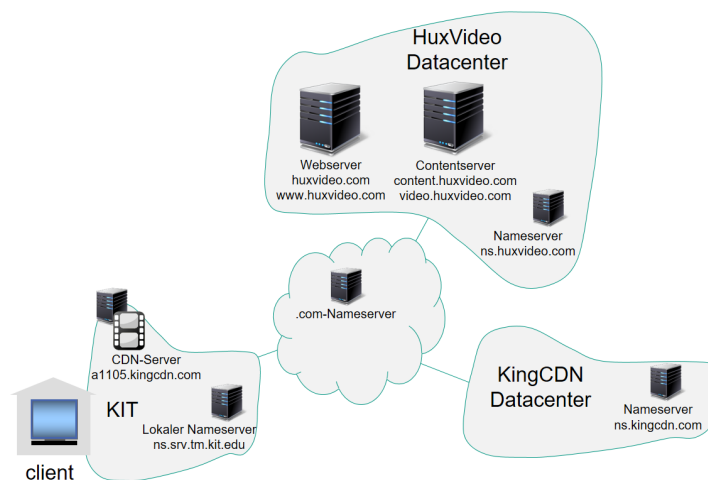


Abbildung 1: HuxVideo-Rechenzentrum, Client, und KingCDN

- (a) Zunächst müssen Sie Ihren Name-Server konfigurieren. Welche Informationen müssen Sie eintragen?
- (b) Im nächsten Schritt müssen Sie Ihrem Domain-Registrar den Name-Server mitteilen. Welche Informationen müssen Sie dazu übergeben? Müssen Sie noch etwas beachten?

- (c) Sie haben Ihren Server fertig implementiert, und möchten diesen nun testen. Dazu rufen Sie erstmalig die Website `www.huxvideo.com` aus dem Netzwerk des KIT auf. Der Name-Server des KIT kennt bereits einen autoritativen Name-Server für .com-TLDs. Halten Sie in einem Weg-Zeit-Diagramm den Ablauf aller DNS- und HTTP-Nachrichten fest, die notwendig sind, bis Ihnen die Startseite angezeigt wird, und beschreiben Sie die einzelnen Schritte.
- (d) Was würde sich ändern, wenn Sie auf den Name-Server des KIT verzichten, und die Namensauflösung stattdessen iterativ von Ihrem Rechner geschehen würde? *Hinweis:* Da Sie vorher auf `snapchat.com` gesurft sind, kennt Ihr Rechner ebenfalls den .com-Name-Server.
- (e) Testweise wählen Sie nun ein Video aus Ihrem Angebot aus. In Ihrem Webbrowser öffnet sich nun der Videoplayer, welcher die ID des abzuspielenden Videos kennt. Wie gelangt der Videoplayer an das Video (Es gibt zwei mögliche Lösungen; entscheiden Sie sich für eine)? Halten Sie den Verlauf der notwendigen DNS- und HTTP-Anfragen in einem Weg-Zeit-Diagramm fest.
- (f) Ihre Videoplattform ist erfolgreich und wächst beständig. Allmählich beginnen Ihre Kunden, sich über lange Ladezeiten und mangelnde Bandbreite zu beschweren, weshalb Sie sich entscheiden, den Anbieter *KingCDN* damit zu beauftragen, Ihre Inhalte auszuliefern. Um weiterhin ein wenig Kontrolle zu behalten, stellen Sie sicher, dass die Manifest-Dateien sich auf Ihrem eigenen Contentserver unter `content.huxvideo.com` befinden. Die Manifestdateien adressieren Chunks auf `video.huxvideo.com`. Welche Schritte sind notwendig, um Ihre Videos auf den Server des CDN-Anbieters umzuziehen?
- (g) Das neue Setup testen Sie wieder aus dem Netz des KIT heraus. Da Studenten und Mitarbeiter normalerweise keine Videos am Campus ansehen, kennt der Name-Server des KIT Ihre Videoplattform noch nicht. Außerdem nutzen Sie einen fremden Rechner, um Caching-Problemen vorzubeugen. Halten Sie in Weg-Zeit-Diagrammen den Ablauf aller DNS- und HTTP-Nachrichten fest, die notwendig sind, wenn Sie
 - Die Startseite von `www.huxvideo.com` aufrufen.
 - Ein Video ansehen.

Aufgabe 4: SMTP in der Praxis

Hinweis: Die so versendeten E-Mails werden möglicherweise als Spam eingestuft. Schauen Sie also in Ihrem Spam-Ordner nach!

- (a) Ermitteln Sie zunächst den *fully-qualified domain name* Ihres Rechners. Führen Sie dazu einen Reverse-lookup Ihrer (öffentlichen) IP-Adresse durch (`dig @server -x youripaddress`). Diesen benötigen Sie später, um sich beim E-Mail-Server zu identifizieren.
- (b) Als nächstes benötigen Sie den E-Mail-Server des Empfängers. Ermitteln Sie dazu den MX-Record der Domain in der E-Mail-Adresse.
- (c) Nun können Sie sich via Telnet mit dem E-Mail-Server des Empfängers verbinden. Nutzen Sie SMTP zur Übergabe der Nachricht.

- (d) In der Vorlesung wurden erweiterte Optionen, wie zum Beispiel der Betreff oder CC, ignoriert. Informieren Sie sich in RFC 822 darüber, wie Sie Ihrer E-Mail einen Betreff hinzufügen können. Schicken Sie nun eine neue E-Mail mit Betreff!

Aufgabe 5: Unterschiede im Nachrichtenaustausch

In den folgenden Szenarien sendet Alice jeweils eine Nachricht an Bob. Beide Nutzer befinden sich dabei in einem Hörsaal des KITs und sind zurzeit mit dem WKIT verbunden. Beschreiben Sie für jedes Szenario, wie die Nachrichtenübertragung von Sender zu Empfänger abläuft (beispielsweise welche Schritte notwendig sind, welche Zwischensysteme eventuell involviert sind, welche Protokolle zwischen den einzelnen Systemen gesprochen werden, ...). Erläutern Sie weiterhin, was *Nutzer-zu-Nutzer*- und *Ende-zu-Ende*-Kommunikation in diesen Szenarien bedeutet.

- (a) Nachrichtenaustausch via WhatsApp?
- (b) Nachrichtenaustausch per Mail über die KIT-Mailadressen?
- (c) Nachrichtenaustausch per Mail von einer KIT-Mailadresse an eine web.de-Mailadresse?
- (d) Nachrichtenaustausch über den XMPP-Server des KITs?

Welche Fehler können bei der Nachrichtenübertragung auftreten? Welche Auswirkungen haben diese?

Aufgabe 6: Telnet und NetCat

Zum senden von Nachrichten über TCP gibt es neben Telnet auch das Tool NetCat. Das Kommando dafür ist `nc`, z.B.: `nc i72tmdjango.tm.kit.edu 8000`.

Hinweis für Windows-Nutzer: Viele NetCat-Ports für Windows haben Probleme mit Antivirensoftware, da manche Funktionen des Tools prinzipiell illegal sein können (z.B. Port-Scanning). Wir empfehlen deshalb, Linux für die Lösung der entsprechenden Aufgaben einzusetzen.

- (a) Gibt es Unterschiede zwischen den Tools Telnet und NetCat? Welche?
- (b) Nutzen Sie NetCat, um eine HTTP-Anfrage für `http://www.kit.edu` zu senden. Wieso erhalten Sie einen Fehler? Lässt sich der Fehler umgehen? Wieso kann `i72tmdjango.tm.kit.edu` eine Anfrage mit NetCat trotzdem verarbeiten?

Aufgabe 7: Präsenz: Socket-Tutorial

In der Übung am 17. Mai werden wir die Grundlagen zu Sockets besprechen - dies ist die Programmierschnittstelle zur Transportschicht. Wenn Sie sich vorab zu dem Thema informieren wollen, können Sie zum Beispiel hier lernen, in Java über TCP zu kommunizieren: <https://docs.oracle.com/javase/tutorial/networking/sockets/index.html>

Aufgabe 8: HTTP über TCP-Sockets

Hinweis: In der Übung am 17. Mai werden wir die Grundlagen zu Sockets besprechen. Die Lösung zu dieser Aufgabe wird in der dritten Übungsstunde vorgestellt. Sie können sich mit der Bearbeitung dieser Aufgabe also länger Zeit lassen.

Auf dem 1. Übungsblatt wurde ein einfacher, HTTP-basierter Chatserver vorgestellt: Die URL `http://i72tmdjango.tm.kit.edu:8000/chat` akzeptiert sowohl GET- als auch POST-Anfragen. In einer POST-Anfrage sollten die Parameter `user` und `message` vorhanden sein, sodass man eine Chat-Nachricht an den Gruppenchat als der angegebene User senden kann. Ferner lehnt dieser Server jede HTTP-Anfrage, die einen User-Agent beinhaltet, ab.

Schreiben Sie ein Clientprogramm für diesen Chatserver in einer Programmiersprache Ihrer Wahl. Dem Benutzer sollen die Nachrichten im Gruppenchat angezeigt werden, und man soll eigene Nachrichten versenden können. Es bleibt Ihnen überlassen, ob das Programm ein Kommandozeilen-Tool ist, ein textbasiertes Programm, oder ein Programm mit graphischer Oberfläche.

Nutzen Sie zur Netzkommunikation die Socket-API, und senden Sie die Nachrichten über TCP-Sockets. Verzichten Sie dabei auf fertige HTTP-Bibliotheken!

Es ist empfehlenswert, zuerst Erfahrung mit der GET-Methode zu sammeln, indem Sie zunächst nur die Lesefunktion implementieren. Lassen Sie dabei Raum für eine Schreibfunktionalität, die sie im Anschluss nachrüsten können. Schreiben Sie dazu eine Funktion, welche einen beliebigen String als Request versenden und die Antwort des Servers zurückgeben kann, und übergeben Sie dieser Funktion wahlweise den entsprechenden GET- oder POST-Request.

Hinweis: Seien Sie fair zu Ihren Kommilitonen und überlasten Sie nicht den Server.